

FRONT

BACK

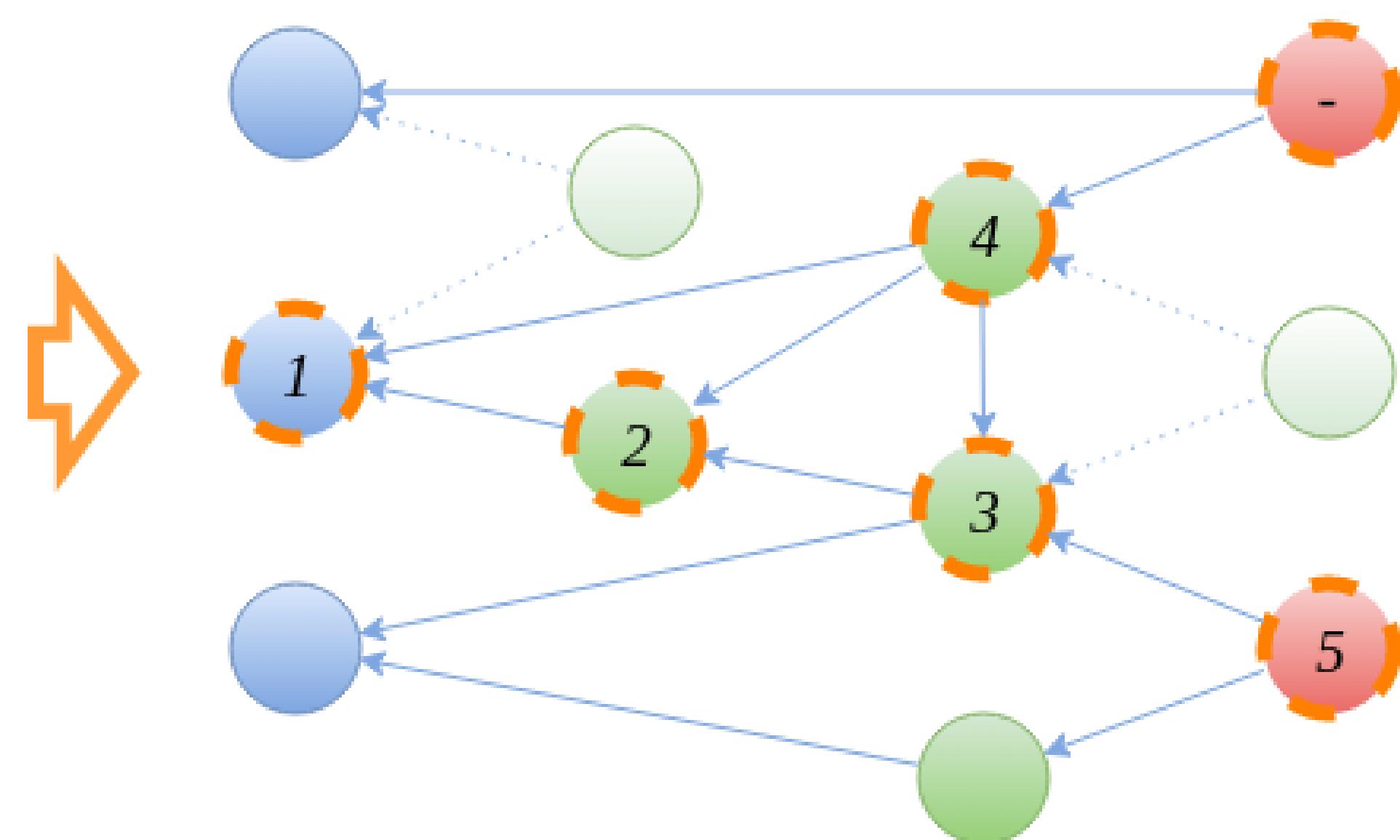
LEFT

RIGHT



Simple and performant
reactivity
for building
user interfaces.

Fine-Grained Reactivity



```
import { onCleanup, createSignal } from "solid-js";

const CountingComponent = () => {
    const [count, setCount] = createSignal(0);

    const interval = setInterval(
        () => setCount(count => count + 1),
        1000,
    );

    onCleanup(() => clearInterval(interval));

    return <div>Count value is {count()}</div>;
};
```

```
import { onCleanup, createSignal } from "solid-js";

const CountingComponent = () => {
    const [count, setCount] = createSignal(0);

    const interval = setInterval(
        () => setCount(count => count + 1),
        1000,
    );

    onCleanup(() => clearInterval(interval));

    return <div>Count value is {count()}</div>;
};
```

```
import { onCleanup, createSignal } from "solid-js"; > rendered!

const CountingComponent = () => {
  const [count, setCount] = createSignal(0);

  const interval = setInterval(
    () => setCount(count => count + 1),
    1000,
  );

  onCleanup(() => clearInterval(interval));

  console.log("rendered!");

  return <div>Count value is {count()}</div>;
};
```

```
import { useEffect, useState } from "react";

const CountingComponent = () => {
  const [count, setCount] = useState(0);

  useEffect(() => {
    const interval = setInterval(
      () => setCount(count => count + 1),
      1000,
    );

    return () => clearInterval(interval);
  }, []);

  console.log("rendered!");

  return <div>Count value is {count}</div>;
};
```

> rendered!
> rendered!
> rendered!
> rendered!
> rendered!
> rendered!
...

State, Effect⁺

Reactivity, Signal

함수형 컴포넌트

사라지는 컴포넌트

JSX

Fine-Grained

Solid.js feels like
what I always wanted
React to be